Revision date: 09/18/2018

**COMS 363 Section C**
**Assignment III**
**Fall 2018**

**Date assigned:** 9/18/2018
**Due date:** 9/24/2018 by 11:50pm
**Percentage in your final grade:** 5%
**Maximum score for the assignment:** 100 points

**Objectives**

1. Practice query languages for relational and graph database management systems.

**Instructions:**

The assignment is individual work, not a group work. Compress all the answers into one zip file. Name the file <netid>HW3.zip where <netid> is replaced by your netid. Do all your homework on your virtual machine.

**Software requirements:**

- Neo4j Community Windows version 3.2.3 installed on your virtual machine
- MYSQL Community Server 8.0 and MySQL Workbench installed on your virtual machine

**Questions:**

1. (50 points) Consider the following relational schemas from homework 2.

   Suppliers(sid: int, sname: VARCHAR(30), address: VARCHAR(50))
   Parts(pid: int, pname: VARCHAR(30), color: VARCHAR(10))
   Catalog(sid: int, pid: int, cost: double)

   Write SQL queries to find the information below. Put your queries in <netid>HW3_Q1.sql and capture screenshots after executing the queries. Submit both <netid>HW3_Q1.sql and the screenshots of all the queries. Replace <netid> with your actual netid.

   a) (3 points) Return the number of black parts.
   b) (3 points) List snames of all suppliers who supply any black part.
   c) (4 points) Find snames of suppliers who supply every black part.
      **Hint:** You may use the answer of a) as the subquery in **having clause.**
   d) (10 points) List pids, pnames, and the minimum cost for that part among all the suppliers supplying the parts in descending order of the minimum cost. **Hint:** Use **group by clause.**
   e) (10 points) Add a new supplier with sid=17, sname='Pak', and address='Iowa State University' in the Suppliers table.
   f) (10 points) Show sids and snames of suppliers who do not supply any part.
   g) (10 points) Write a different SQL query that gives the exact same answer as in the question f)

   **Hint**: For questions e-g, see the class activity questions to get ideas.

2. (5) Unzip tweets.zip file that comes with this assignment. This file contains a Neo4j graph database of tweets of Twitter users of the 2016 presidential candidates, state's senators, state's house

representatives, state's reporters, and state's Senate. Set your Neo4j database location to where the graph.db folder is located after you unzip the file. Figure 1 shows the schema of the graph database. Each rectangle represents a group of nodes with similar properties. We use Neo4j labels to implement the grouping. The node labels are State, User, Century, Year, Month, Day, Tweet, Url, and Hashtag. In the diagram, User (Posted) and User (Mentioned) both have the same User label. The edge labels are FROM, MENTIONED, POSTED, URL_USED, TAGGED, HAS_TWEET, HAS_YEAR, HAS_MONTH, and HAS_DAY. Edges in this database have no attributes.

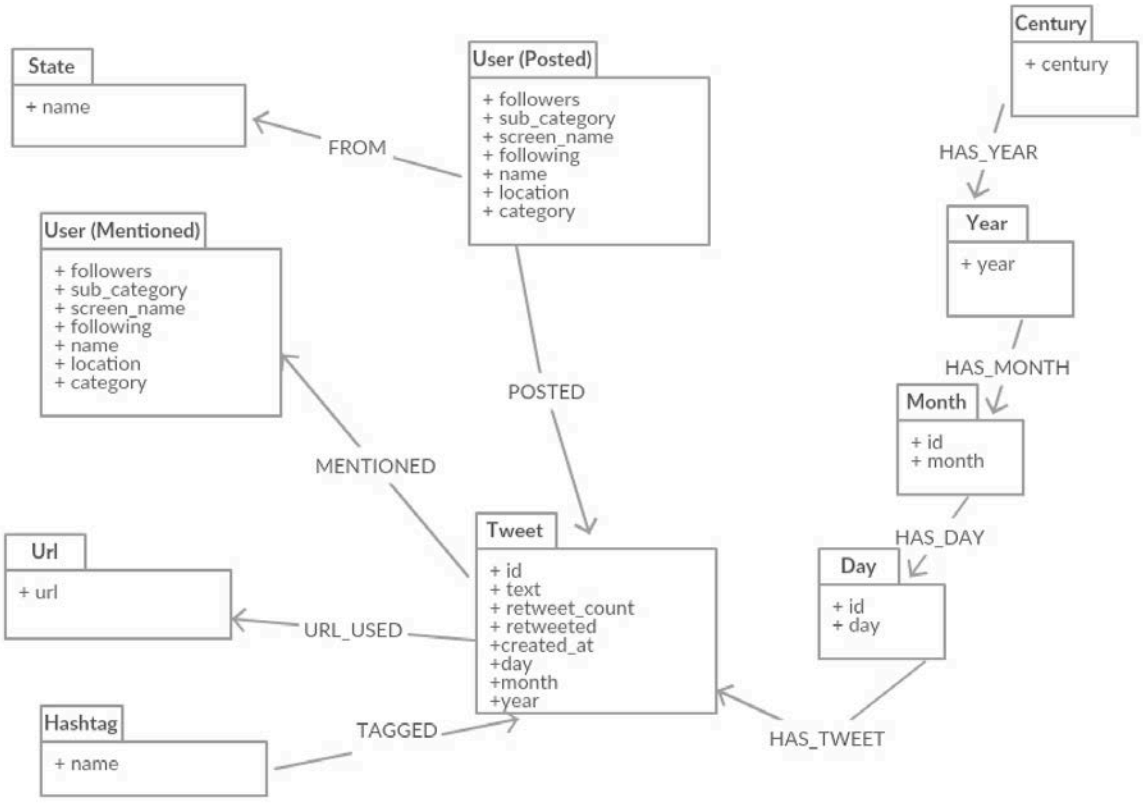


**Figure 1: Schema of the graph database of tweets of state legislatures and presidential candidates for the 2016 presidential election.**

The design of this data model follows the basic graph database design guidelines. Nodes represent entities and an edge represents a relationship between two nodes. Properties of a tweet with possible multiple values such as hashtags and urls are represented as nodes. Tweet nodes have properties: id, retweet_count (the number of retweets of this tweet), retweeted (whether this tweet has been retweeted), tweet text, created_at (of type long representing the timestamp---the number of milliseconds since 1/1/1970---in which the tweet was posted), day, month, and year. User nodes have the following properties: name, screen_name, followers (indicating the number of followers), following (indicating the number of people this user follows), sub_category, category, location, and name. The sub_category indicates the party to which the user belongs: 'GOP', 'Democrat', 'na', or null. The category is either a Senate account (senate_group), presidential_candidate, reporter, Senator, General, or null. The name property can have an empty string as a value.

State nodes have the name property containing the name of the state. There are 52 State nodes with the state of Florida appears twice as 'Florida' and 'FL'; furthermore, there is one 'na' State node for the user without state information such as presidential candidates. Hashtag nodes have the tag name property. Url has the url property that can also be an empty string.

This model also uses a time tree that was created to support the time range queries. The century node has the property century with an integer value of 21 to represent the current century. Each year node has edges only to the month nodes of that year. Each month node has edges only to the day nodes of that month only. In other words, the day nodes are not shared across multiple months. A day node has edges to tweets posted on that particular day of that particular month, year and century.

3 (45 points) Provide the Cypher query for each of the following questions. The screenshot for each query must show the query and the result. Put all the queries in a text file and name it <netid>HW3_Q3.cypher. Submit both the Cypher file and the screenshots.

**Note**: The function that converts the attribute of type text to integer is toInteger(<attribute name>) where <attribute name> is replaced by the actual attribute name.

a. (9 points) Find top 10 tweets (in the descending order of the retweet count) posted during the month of Jan. 2016. In the reverse order of the retweet count, display the retweet count, the tweet text, the name and the sub_category of the user who posted these tweets.
   **Note:** Properties retweet_count, day, month, year are of type text.
b. (9 points) Same as Q1, but write the query using the time tree instead. The property century of the Century node is of integer type. The property of day, month, and year are of text type.
c. (9 points) Find 20 distinct users whose state is not 'na' and have used at least one of the hashtags in this set {GOPDebate, DemDate, GOP} in any of their tweets; show the name of the user and the state to which the user belongs.
d. (9 points) Find top 15 users with the most number of followers; show the name, the category, and the number of followers of these users in descending order of the number of followers.
e. (9 points) Show the name of the user and the list of non-empty urls in the tweets posted by the user in the category 'reporter' on Jan. 2, 2016. Limit to 20 names.
   **Hint**: Use collect(distinct <attributename>) to put all the values of the attribute as a list.